Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently Amended) A microcontroller the programming of which is carried out in at least one machine-dependent assembler language in which the assembler commands, with the exception of conditional program jumps or program branches, respectively, can be executed in essence independently of data, characterized by at least one random number generator assigned to the microcontroller ~~can be executed~~, by means of which the program jumps or program branches can be executed in dependence on the state of the random number generator and/or independently of the internal state of the programming of the microcontroller.

2. (Previously Presented) A microcontroller as claimed in claim 1, characterized by at least one, in particular bit-addressable, random number register assigned to the random number generator.

3. (Previously Presented) A microcontroller as claimed in claim 1, characterized by an embodiment as a smartcard controller.

4. (Previously Presented) An electrical or electronic device controlled by means of at least one microcontroller as claimed in claim 1.

5. (Previously Presented) A method for processing the programming of a microcontroller executed in at least one machine-dependent assembler language, the assembler commands, with the exception of conditional program jumps or branches, being executed essentially independently of data, characterized in that the program jumps or program branches are executed in dependence on the state of at least one random number generator and/or independently of the internal state of the programming of the microcontroller.

6. (Previously Presented) A method as claimed in claim 5, characterized in that the random number generated by the random number generator read via software via registers and the random number read is then evaluated with a conditional program jump or branch.

7. (Previously Presented) A method as claimed in claim 5, characterized in that, if at least one, in particular bit-addressable, random number register is present, testing per bit of the random number register and a conditional jump or branch is carried out.

8. (Previously Presented) A method as claimed in claim 5, characterized by the implementation of at least one assembler command ("branch on random bit"), a defined bit of the

random number register being supplied, in particular directly, to the condition input for the conditional jump or branch.

9. (Previously Presented) A method as claimed in claim 5, characterized in that at least one Arithmetic Logic Unit (ALU) flag controlling the conditional jumps or branches is replaced, in particular via the software, by at least one bit of the random number register, so that the conditional jumps or branches corresponding to the bit of the Arithmetic Logic Unit are controlled by the bit of the Random Number Register.

10. (Currently Amended) A use of a microcontroller as claimed in ~~at least one of claims 1 to 3 and/or of a method as claimed in claim 5~~ claim 1 for completely concealing the programming running on the microcontroller, so that at least one program running on the microcontroller is unpredictable and non-reproducible for an external observer.

11. (New) A microcontroller comprising:

a central processing unit;

a memory accessible to the central processing unit, wherein the memory comprises instructions and wherein the central processing unit is configured for:

accessing the instructions, wherein the instructions comprise instruction sequences and where each instruction sequence produces a same result value for a same input value;

receiving a random number associated with an instruction sequence;

receiving the same input value; and

executing the instruction sequence associated with the random number using the same input value to produce the same result.

12. (New) The microcontroller of claim 1, wherein the instructions further comprise a jump instruction subject to a condition which if true directs the central processing unit to a program address, wherein the instruction sequence is located at the program address, and wherein the random number is associated with the instruction sequence via the program address.

13. (New) The microcontroller of claim 1, wherein the instructions further comprise a jump instruction subject to a condition which if true directs the central processing unit to a program address, wherein the instruction sequence is located at the program address, and wherein the random number is associated with the instruction sequence via the condition.

14. (New) The microcontroller of claim 1, wherein the random number is generated by a random number generator assigned to the microcontroller.